

Non-parametric Inference and Coordination for Distributed Robotics

Brian J. Julian^{*†}, Michael Angermann[‡], and Daniela Rus^{*}

Abstract—This paper presents non-parametric methods to infer the state of an environment by distributively controlling robots equipped with sensors. Each robot represents its belief of the environment state with a weighted sample set, which is used to draw likely observations to approximate the gradient of mutual information. The gradient leads to a novel distributed controller that continuously moves the robots to maximize the informativeness of the next joint observation, which is then used to update the weighted sample set via a sequential Bayesian filter. The incorporated non-parametric methods are able to robustly represent the environment state and robots' observations even when they are modeled as continuous-valued random variables having complicated multimodal distributions. In addition, a consensus-based algorithm allows for the distributed approximation of the joint measurement probabilities, where these approximations provably converge to the true probabilities even when the number of robots, the maximum in/out degree, and the network diameter are unknown. The approach is implemented for five quadrotor flying robots deployed over a large outdoor environment, and the results of two separate exploration tasks are discussed.

I. INTRODUCTION

We address the problem of using many collaborating robots equipped with sensors to acquire information from a large-scale environment. The key to our solution is in the use of non-parametric¹, sample-based representations of the probability distributions present in the system. Computations on these non-parametric representations are performed using a novel consensus-based algorithm. This approach leads to scalability with respect to the number of robots, and allows for completely decentralized computation under few

This work is sponsored by the Department of the Air Force under Air Force contract number FA8721-05-C-0002. The opinions, interpretations, recommendations, and conclusions are those of the authors and are not necessarily endorsed by the United States Government. This work is also supported in part by ONR grant number N00014-09-1-1051, NSF grant number EFRI-0735953, MIT Lincoln Laboratory, and the Boeing Company.

The authors would like to thank Emilio Frazzoli, Mac Schwager, and Patrick Robertson for their many significant contributions in the areas of consensus algorithms, information theory, and probabilistic methods. The authors would also like to thank Martin Frassl, Michael Lichtenstern, Michael Walter, Ulrich Epple, and Frank Schubert for their assistance with the experiments. Lastly, the authors would like to thank the NASA World Wind Project for providing the World Wind technology, which has been an essential part for visualization.

^{*}Brian J. Julian and Daniela Rus are with the Computer Science and Artificial Intelligence Lab, MIT, Cambridge, MA 02139, USA, bjulian@mit.edu and rus@csail.mit.edu

[†]Brian J. Julian is also with MIT Lincoln Laboratory, 244 Wood Street, Lexington, MA 02420, USA

[‡]Michael Angermann is with the Institute of Communications and Navigation, DLR Oberpfaffenhofen, 82234 Wessling, Germany, michael.angermann@dlr.de

¹We use the term *non-parametric* to convey that we do not assume that the statistics of the involved random variables can be exactly described by particular distributions with finite numbers of parameters.

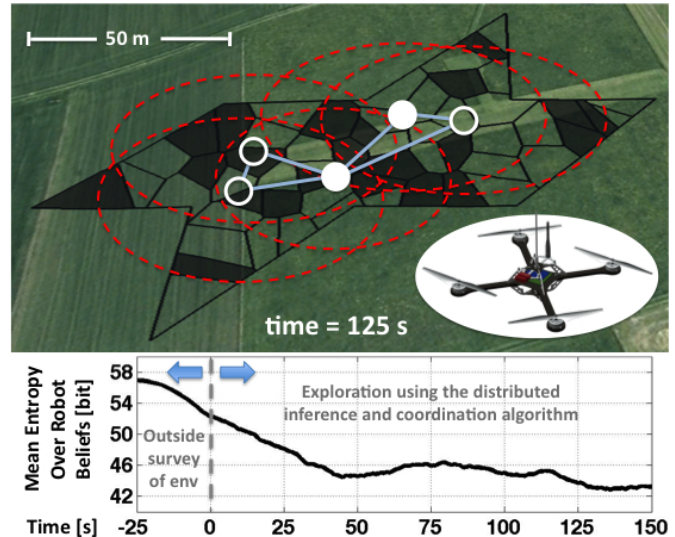


Fig. 1. Five quadrotor flying robots (white circles) are tasked to explore a 150 m wide discretized outdoor environment containing 58 cells of binary state. From time $t \in [-25, 0)$ s, a non-parametric Bayesian filter runs on static robots that partially observe the environment from outside its boundary. Exploration by the robotic sensor network (blue lines) starting at time $t = 0$ s is accomplished by a gradient-based distributed controller that continuously moves the robot sensors (dashed red circles) to minimize the uncertainty of the non-parametric inference. In parallel, several robots are assigned by a high-level communication scheme to act as dynamic communication relays (white filled circles). The result is a steady decrease in average entropy of each robot's belief of the environment state, as shown in the lower plot.

probabilistic assumptions. Specifically, we do not assume that any probability distribution can be accurately represented by a Gaussian distribution. Assuming Gaussianity significantly simplifies many aspects of the system. However, Gaussian distributions often do not adequately represent the characteristics of realistic environments and sensors, and may result in misleading inferences and poor controller performance. Instead, we approximate the robots' beliefs and likely observations using sample sets that are distributively formed and provably unbiased [12]. These nonparametric sampled distributions are used both for the sequential Bayesian filter to update the robots' beliefs, and for the information seeking controller to move the robots and orient their sensors.

The robots are controlled to seek informative observations by moving along the gradient of mutual information at each time step. However, the computation of the mutual information gradient and the sequential Bayesian filter require global knowledge that is not readily available in a decentralized setting. To overcome this requirement, we use a consensus-based algorithm specifically designed to

successively approximate the required global quantities with local estimates. These approximations provably converge to the desired global quantities as the size of the consensus rounds grows or as the network graph becomes complete. Convergence is guaranteed for robots without any knowledge of the number of the robots in the network, nor any knowledge about the corresponding graph's topology (e.g., maximum in/out degree).

In our previous work [11], we only considered environment states that could be represented by discrete-valued random variables. Here, this simplifying assumption is not needed as our non-parametric methods are able to properly model continuous-valued random variables. In addition, we further distinguish the presented approach by not requiring the robots to know aspects of the global network topology, whether it be the number of robots, the maximum in/out degree, or the network diameter. This capability is made possible by using a FloodMax algorithm in parallel with the consensus algorithm and showing that convergence to the true joint measurement probabilities is preserved. The end result is a distributed inference and coordination algorithm that makes few assumptions about the underlying probability distributions, significantly relaxes networking requirements, and is inherently scalable.

Non-parametric filters have a long standing history in Bayesian estimation, and have recently become popular in robotics as the platforms become more computationally capable. In an early work, Engelson and McDermott [5] used a sequential Monte Carlo method to construct a mapping algorithm robust enough to address the kidnapped robot problem. Since then, non-parametric algorithms have become commonplace in localization [1], simultaneous localization and mapping [15], and target tracking [18]. Fox et al. [6] applied these algorithms to multiple collaborating robots using a sample-based version of Markov localization. In addition, Belief Propagation [17] has seen non-parametric extensions [10], which use mixtures of Gaussians to solve graphical inference problems.

With respect to multi-robot coordination for information acquisition tasks, most works have assumed linear Gaussian systems. For example, Lynch et al. [13] proposed using a consensus based Kalman filter to distributively control a team of robots to decrease the error variance of the state estimate. In addition, Cortés [4] developed a distributed Kriged Kalman filter for estimating environmental fields, which enabled a gradient ascent strategy. A notable exception to the standard Gaussian approaches concerns recent efforts by Hoffmann and Tomlin [9], who proposed a sequential Monte Carlo method to propagate a Bayesian estimate, then used greedy and pair-wise approximations to calculate mutual information. The use of mutual information to formulate multi-robot controllers follows a long lineage of information theoretic approaches [3], [2], [8], however, only recently has the analytically derived expression for the gradient of mutual information been used for control [11], [19].

The paper is organized as follows. In Section II we

formulate the general setup for inferring the environment state by controlling multiple robots equipped with sensors, then expand this concept to decentralized systems. In Section III we introduce non-parametric methods for representing the robots' beliefs, likely observations, and measurement probabilities for inference and coordination. In Section IV we show that convergence of a consensus-based algorithm that distributively approximates the joint measurement probabilities can be achieved in networks of unknown topology. Finally, we demonstrate the non-parametric methods in Section V through hardware experiments with five quadrotor flying robots.

II. PROBLEM FORMULATION

We state the basic definitions and results for distributed inference and coordination that will be used in this paper. A more thorough discussion can be found in [11].

We wish to infer the state of a bounded environment, \mathcal{Q} , from measurements obtained by a number of robots equipped with sensors. Let the environment state at time step k be represented by a discrete-time random variable, X_k , that takes values from an alphabet, \mathcal{X} , at constant time intervals, T_s . In our work to enable a multi-robot system to rapidly assess the state of disaster-affected environments, the environment state can represent a wide spectrum of relevant information, ranging from the presence of fires and harmful substances to the structural integrity of buildings.

Let there be n_r robots, where each robot moves in a configuration space, $\mathcal{C}^{[i]} = \mathbb{R}^{r_q} \times \mathbb{S}^{s_q} \supset \mathcal{Q}$, with \mathbb{R}^{r_q} and \mathbb{S}^{s_q} representing the r_q -dimensional Euclidean space and the s_q -dimensional sphere, respectively. The configuration space describes the position of an individual robot and the orientation of its sensors. The Cartesian product of the configuration spaces, $\mathcal{C} = \{\mathbb{R}^{r_q} \times \mathbb{S}^{s_q}\}^{n_r}$, represents the configuration space for the system of robots, from which the n_r -tuple $c_t = (c_t^{[1]}, \dots, c_t^{[n_r]})$ denotes the system's configuration at continuous time $t \geq 0$.

Each robot cannot perfectly measure the environment state, but instead makes an observation with its sensors that are influenced by noise. The robots' synchronous observations together form a joint observation, which we model as an n_r -tuple random variable, $Y_k = (Y_k^{[1]}, \dots, Y_k^{[n_r]})$, that takes values, $y = (y^{[1]}, \dots, y^{[n_r]})$, from an alphabet, $\mathcal{Y} = \prod_{i=1}^{n_r} \mathcal{Y}^{[i]}$. The relationship between the environment state and the noisy observation is described by joint measurement probabilities, $\mathbb{P}(Y_k|X_k)$, which are functions of the system's configuration. We assume conditional independence of the robots' observations, giving $\mathbb{P}(Y_k|X_k) = \prod_{i=1}^{n_r} \mathbb{P}(Y_k^{[i]}|X_k)$. Note that the alphabet representing the environment state alphabet is known among all robots. Thus from Bayes' Rule, we can use a joint observation and the system's prior distribution, $\mathbb{P}(X_k)$, to compute the posterior distribution,

$$\mathbb{P}(X_k|Y_k) = \frac{\mathbb{P}(X_k) \prod_{i=1}^{n_r} \mathbb{P}(Y_k^{[i]}|X_k)}{\int_{x \in \mathcal{X}} \mathbb{P}(X_k=x) \prod_{i=1}^{n_r} \mathbb{P}(Y_k^{[i]}|X_k=x) dx} \quad (1)$$

At any given time, a robot can choose a control action, $u_t^{[i]}$, from its control space, $\mathcal{U}^{[i]} \subset \mathcal{C}^{[i]}$. We model the

robot as having continuous-time integrator dynamics, $\frac{dc_t^{[i]}}{dt} = u_t^{[i]}$, which is a common assumption in the multi-robot coordination literature. Since our objective is to best infer the environment state, we are motivated to move the robots into a configuration that minimizes the expected uncertainty of the inference after receiving the next joint observation. Our optimization objective is equivalent to maximizing the mutual information between the environment state and the joint observation,

$$U_k := I(X_k, Y_k) = \int_{y \in \mathcal{Y}} \int_{x \in \mathcal{X}} \mathbb{P}(X_k = x) \times \prod_{i=1}^{n_r} \mathbb{P}(Y_k^{[i]} = y^{[i]} | X_k = x) \log \left(\frac{\mathbb{P}(X_k = x | Y_k = y)}{\mathbb{P}(X_k = x)} \right) dx dy, \quad (2)$$

which represents the utility function for the system with log denoting the natural logarithm. We are interested in gradient ascent approaches using the partial derivative of the utility function with respect to the configuration of a robot,

$$\frac{\partial U_k}{\partial c_t^{[i]}} = \int_{y \in \mathcal{Y}} \int_{x \in \mathcal{X}} \frac{\partial \mathbb{P}(Y_k^{[i]} = y^{[i]} | X_k = x)}{\partial c_t^{[i]}} \mathbb{P}(X_k = x) \times \prod_{v \neq i} \mathbb{P}(Y_k^{[v]} = y^{[v]} | X_k = x) \log \left(\frac{\mathbb{P}(X_k = x | Y_k = y)}{\mathbb{P}(X_k = x)} \right) dx dy. \quad (3)$$

By incorporating sampling (for scalability, see Section III) and consensus (for decentralization, see Section IV) for a robot network modeled as an undirected graph \mathcal{G}_k , we formalized a gradient-based distributed controller that follows the approximated gradient of mutual information. By following this gradient, the robots better position themselves for the next joint observation, which is then used for a sequential Bayesian filter to distributively update each robot's belief of the environment state. The proposed controller runs in constant time with respect to the number of robots, and is provably convergent between consensus rounds and, under certain conditions, locally optimal.

III. NON-PARAMETRIC METHODS

We begin by presenting the high-level architecture of the distributed inference and coordination algorithm in Algorithm 1, then discuss the details of its non-parametric methods.

A. Environment State and Observation Sample Sets

Let each robot maintain a weighted environment state sample set, $\tilde{\mathcal{X}}_k^{[i]} = \{(\tilde{x}_k^{[i,j]}, \tilde{w}_k^{[i,j]}) : j \in \{1, \dots, n_x\}\}$, of size n_x , where each sample, $\tilde{x}_k^{[i,j]} \in \mathcal{X}$, has a corresponding weight, $\tilde{w}_k^{[i,j]} \in (0, 1)$. Each sample is a candidate instantiation of the environment state, and the pairing of the samples and their corresponding weights represents a non-parametric approximation of the robot's belief of the environment state.

Using this set, samples of likely observations for each robot are formed. Let each robot create a temporary unweighted environment state sample set by drawing n_y samples from the weighted sample set with probabilities proportional to the corresponding weights. Note that the drawn samples represent equally likely state instantiations (they are formed in a method analogous to the *importance sampling*

Algorithm 1 Distributed Inference and Coordination()

Require: The i th robot knows its configuration, its measurement probabilities, and the extent of the environment.

- 1: Initialize the weighted environment state sample set from Section III-A.
 - 2: **loop**
 - 3: Distributively approximate the sampled joint measurement probabilities using `Belief_Consensus(sampled)` from Section IV-B.
 - 4: Apply controller from Corollary 1 in Section III-B.
 - 5: Distributively approximate the joint measurement probabilities using `Belief_Consensus(observed)` from Section IV-B.
 - 6: Update weighted environment state sample set using `Sequential_Importance_Resampling()` from Section III-C.
 - 7: **end loop**
-

step for particle filters [20]). A robot's observation sample set, $\tilde{\mathcal{Y}}_k^{[i]} = \{\tilde{y}_k^{[i,\ell]} : \ell \in \{1, \dots, n_y\}\}$, is then formed by drawing one observation sample for each entry in the temporary state sample set using the robot's measurement probabilities. The corresponding sampled measurement probabilities become

$$\mathbb{P}(\tilde{Y}_k^{[i]} | X_k) = \frac{\mathbb{P}(Y_k^{[i]} | X_k)}{\sum_{\ell=1}^{n_y} \mathbb{P}(Y_k^{[i]} = \tilde{y}_k^{[i,\ell]} | X_k)},$$

where $\tilde{Y}_k^{[i]}$ is a random variable that takes values from $\tilde{\mathcal{Y}}_k^{[i]}$.

We then define the joint observation sample set, $\tilde{\mathcal{Y}}_k$, as the unweighted set of n_r -tuples formed from the robots' observation samples having equal indices. More formally, we have that $\tilde{\mathcal{Y}}_k = \{\tilde{y}_k^{[\ell]} = (\tilde{y}_k^{[1,\ell]}, \dots, \tilde{y}_k^{[n_r,\ell]}) : \tilde{y}_k^{[i,\ell]} \in \tilde{\mathcal{Y}}_k^{[i]}\}$. Note that a generic formulation of a joint observation sample set would be the Cartesian product of all the robots' observation sample sets, $\prod_{i=1}^{n_r} \tilde{\mathcal{Y}}_k^{[i]}$, which in size scales exponentially with respect to the number of robots. Here we use the fact that a robot's observation sample set is both unweighted (i.e., all samples are equally likely) and conditionally independent to form an unbiased joint observation sample set of constant size with respect to the number of robots. In other words, each robot independently draws its own observation samples using its local measurement probabilities, and due to conditional independence the concatenation of these samples across all robots is equivalent to a sample set formed by using the system's joint measurement probabilities.

B. Gradient-Based Control

We will show in Section IV-B that by using a consensus-based algorithm, each robot can distributively approximate the sampled joint measurement probabilities, $\mathbb{P}(\tilde{Y}_k | X_k)$, where \tilde{Y}_k is a random variable that takes values from $\tilde{\mathcal{Y}}_k$. Let the matrix $\tilde{p}_k^{[i]}$ denote these approximations (see Section IV-C for details on this matrix), where from Bayes' Rule the

posterior calculation (1) becomes

$$\mathbb{P}(X_k = \tilde{x}_k^{[i,j]} | \tilde{Y}_k = \tilde{y}_k^{[\ell]}) \approx \frac{\tilde{w}_k^{[i,j]} [\tilde{p}_k^{[i]}]_{j\ell}}{\sum_{j'=1}^{n_x} \tilde{w}_k^{[i,j']} [\tilde{p}_k^{[i]}]_{j'\ell}}$$

for all $j \in \{1, \dots, n_x\}$ and $\ell \in \{1, \dots, n_y\}$, with $[\cdot]_{j\ell}$ denoting the matrix entry (j, ℓ) .

By incorporating the weighted environment state sample set, the joint observation sample set, and the sampled joint measurement probability approximations into (2), we define

$$\begin{aligned} \tilde{U}_k^{[i]} &:= \sum_{\ell=1}^{n_y} \sum_{j=1}^{n_x} \mathbb{P}(\tilde{Y}_k^{[i]} = \tilde{y}_k^{[\ell]} | X_k = \tilde{x}_k^{[i,j]}) \\ &\times \frac{\tilde{w}_k^{[i,j]} [\tilde{p}_k^{[i]}]_{j\ell}}{\mathbb{P}(\tilde{Y}_k^{[i]} = \tilde{y}_k^{[\ell]} | X_k = \tilde{x}_k^{[i,j]})|_{t=kT_s}} \log \left(\frac{[\tilde{p}_k^{[i]}]_{j\ell}}{\sum_{j'=1}^{n_x} \tilde{w}_k^{[i,j']} [\tilde{p}_k^{[i]}]_{j'\ell}} \right) \end{aligned} \quad (4)$$

to be the i th robot's approximation of the utility function given its measurement probabilities at time $t = kT_s$. Taking the partial derivative of (4) with respect to the robot's configuration, we have

$$\begin{aligned} \frac{\partial \tilde{U}_k^{[i]}}{\partial c_t^{[i]}} &= \sum_{\ell=1}^{n_y} \sum_{j=1}^{n_x} \frac{\partial \mathbb{P}(\tilde{Y}_k^{[i]} = \tilde{y}_k^{[\ell]} | X_k = \tilde{x}_k^{[i,j]})}{\partial c_t^{[i]}} \\ &\times \frac{\tilde{w}_k^{[i,j]} [\tilde{p}_k^{[i]}]_{j\ell}}{\mathbb{P}(\tilde{Y}_k^{[i]} = \tilde{y}_k^{[\ell]} | X_k = \tilde{x}_k^{[i,j]})|_{t=kT_s}} \log \left(\frac{[\tilde{p}_k^{[i]}]_{j\ell}}{\sum_{j'=1}^{n_x} \tilde{w}_k^{[i,j']} [\tilde{p}_k^{[i]}]_{j'\ell}} \right), \end{aligned} \quad (5)$$

which is a distributed approximation of the gradient of mutual information (3). Multiplying this result by the positive scalar control gain $\gamma^{[i]}$ results in a gradient-based controller that is distributed among the robots and uses the non-parametric representation of the robots' beliefs. We now use LaSalle's Invariance Principle to prove convergence of the controller.

Corollary 1 (Convergence of the Controller). *Let all robots have integrator dynamics, move in the same configuration space, and sense a bounded environment that is a subset of the configuration space. Consider the class of systems where for all robots $i \in \{1, \dots, n_r\}$, the change in measurement probabilities is continuous on the robot's configuration space and equal to zero for all configurations greater than a certain distance away from the environment (e.g., sensors of limited range). Then for a positive scalar $\gamma^{[i]}$, the controller $u_t^{[i]} = \gamma^{[i]} \frac{\partial \tilde{U}_k^{[i]}}{\partial c_t^{[i]}}$ is convergent to zero between the consensus updates of the distributively approximated sampled joint measurement probabilities.*

Proof (Corollary 1). *The proof directly follows the convergence proof for Theorem 3 in [11], using the Lyapunov-type function candidate $V_k = -\sum_{i=1}^{n_r} \tilde{U}_k^{[i]}$.* \square

Remark 1 (Computational Tractability). *The utility gradient approximation (5) requires $\mathcal{O}(n_x n_y)$ time and $\mathcal{O}(n_y)$ memory, where the memory requirement is due to precalculating the summation in the logarithm function for all joint observation samples. Hence, the gradient-based controller scales linearly with respect to the sizes of the environment state and joint observation sample sets. Moreover, computational complexity remains constant with respect to the number of robots.*

C. Sequential Importance Resampling

By following the approximate gradient of mutual information, the robots better position themselves for the next joint observation, $y_k \in \mathcal{Y}$. Once received, an approximation for the joint measurement probabilities, $\mathbb{P}(Y_k = y_k | X_k)$, is distributively calculated by again using a consensus-based algorithm. Let the column vector $p_k^{[i]}$ denote these approximations (again see Section IV-C for details on this vector), where the posterior calculation now becomes

$$\mathbb{P}(X_k = \tilde{x}_k^{[i,j]} | Y_k = y_k) \approx \frac{\tilde{w}_k^{[i,j]} [p_k^{[i]}]_j}{\sum_{j'=1}^{n_x} \tilde{w}_k^{[i,j']} [p_k^{[i]}]_{j'}} \quad (6)$$

for all $j \in \{1, \dots, n_x\}$, with $[\cdot]_j$ denoting the j th row entry. Thus, each robot forms its weighted environment state sample set for the upcoming time step $k+1$ by drawing from its state transition distribution, $\mathbb{P}^{[i]}(X_{k+1} | X_k)$, calculating the corresponding weights from (6), and applying an appropriate resampling technique. The process of updating the weighted sample set is a well-known sequential Monte Carlo method called sequential importance resampling (Algorithm 2).

Algorithm 2 Sequential_Importance_Resampling()

- 1: $\tilde{\mathcal{X}}_{k+1}^{[i]} \leftarrow \emptyset$.
 - 2: **for** $j = 1$ to n_x **do**
 - 3: $\tilde{x}_{k+1}^{[i,j]} \sim \mathbb{P}^{[i]}(X_{k+1} | X_k = \tilde{x}_k^{[i,j]})$.
 - 4: $\tilde{w}_{k+1}^{[i,j]} \leftarrow \frac{\tilde{w}_k^{[i,j]} [p_k^{[i]}]_j}{\sum_{j'=1}^{n_x} \tilde{w}_k^{[i,j']} [p_k^{[i]}]_{j'}}$.
 - 5: $\tilde{\mathcal{X}}_{k+1}^{[i]} \leftarrow \tilde{\mathcal{X}}_{k+1}^{[i]} \cup \{(\tilde{x}_{k+1}^{[i,j]}, \tilde{w}_{k+1}^{[i,j]})\}$.
 - 6: **end for**
 - 7: Apply appropriate resampling technique.
 - 8: **return** $\tilde{\mathcal{X}}_{k+1}^{[i]}$.
-

IV. CONSENSUS ENABLED INFERENCE AND COORDINATION

We now present a novel consensus-based algorithm that is derived the averaging consensus algorithm in [16]. Our algorithm guarantees that all robots' distributed approximations of the joint measurement probabilities converge to the true values when the number of robots, the maximum in/out degree, and the network diameter are unknown. Note that other consensus approaches, including using the network's Metropolis-Hastings weights [21], are also applicable.

A. Maximum In/Out Degree

The FloodMax algorithm is a well studied distributed algorithm used in leader election problems [14]. Traditionally implemented, each robot would transmit the maximum unique identifier (UID) it received up to the given communication round.² After $\text{diam}(\mathcal{G}_k)$ communication rounds, where $\text{diam}(\cdot)$ represents the diameter of a graph, all robots would then know the maximum UID in the network. To solve the

²A communication round, denoted k' , is defined as a single update using a FloodMax algorithm, a consensus algorithm, or both algorithms if run in parallel.

leader elect problem, the robot whose own UID matches the maximum UID of the network would declare itself the leader.

For distributed inference and coordination, the robots do not need to select a leader, but instead need an estimate of the network's maximum in/out degree, Δ_k , that is guaranteed to converge to the true value. Moreover, we assume that the robots only know characteristics that describe their local neighborhood (e.g., number of neighbors). In other words, the robots do not know characteristics describing the overall network topology, such as the number of robots and the network diameter. This restriction implies that the robots may never identify that the maximum in/out degree has been found. Regardless, the robots can still reach an agreement during consensus by using in parallel the FloodMax algorithm described in the following.

Lemma 1. [Maximum In/Out Degree Discovery] *For the connected graph \mathcal{G}_k , consider the following discrete-time time invariant dynamical system,*

$$\delta_{k'+1}^{[i]} = \max \{ \{ \delta_{k'}^{[i]} \} \cup \{ \delta_{k'}^{[v]} : v \in \mathcal{N}_k^{[i]} \} \}, \quad (7)$$

where for each robot $\delta_{k'}^{[i]}$ is initialized to the robot's number of neighbors plus one. Then for all robots after $\text{diam}(\mathcal{G}_k)$ communication rounds, $\delta_{k'}^{[i]}$ is equal to the network's maximum in/out degree plus one.

Proof (Lemma 1). *The proof is a simple extension of the proof for Theorem 4.1 in [14].*

B. Consensus in Networks of Unknown Topology

Consider a system of robots running a discrete-time consensus algorithm [16] of the form

$$\psi_{k'+1}^{[i]} = \psi_{k'}^{[i]} + \epsilon_k \sum_{v \in \mathcal{N}_k^{[i]}} (\psi_{k'}^{[v]} - \psi_{k'}^{[i]}), \quad (8)$$

where $0 < \epsilon_k < 1/\Delta_k$ guarantees that for all robots, $\psi_{k'}^{[i]}$ exponentially converges to the average initial state of all robots, $\sum_{i=1}^{n_r} \psi_0^{[i]}/n_r$. For the robots to select a valid ϵ_k , they need to know either the maximum in/out degree of the network or the number of robots (since $1/n_r < 1/\Delta_k$). Since we are assuming that neither parameter is known, the consensus algorithm is modified to use in parallel the FloodMax algorithm (7). As a result, convergence to the average initial state is preserved as described by the following, and the process of evolving $\delta_{k'}^{[i]}$, $\psi_{k'}^{[i]}$, and $\pi_{k'}^{[i]}$ in parallel over n_π communication rounds will be summarized in Algorithm 3.

Theorem 1 (Convergence of the Consensus Algorithm). *For the connected graph \mathcal{G}_k , consider the following discrete-time time varying dynamical system,*

$$\psi_{k'+1}^{[i]} = \frac{\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}}{\delta_{k'+1}^{[i]}} \psi_0^{[i]} + \frac{\delta_{k'+1}^{[i]} - |\mathcal{N}_k^{[i]}|}{\delta_{k'+1}^{[i]}} \psi_{k'}^{[i]} + \frac{1}{\delta_{k'+1}^{[i]}} \sum_{v \in \mathcal{N}_k^{[i]}} \psi_{k'}^{[v]}, \quad (9)$$

and its exponential form,

$$\pi_{k'+1}^{[i]} = (\pi_0^{[i]})^{\frac{\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}}{\delta_{k'+1}^{[i]}}} (\pi_{k'}^{[i]})^{\frac{\delta_{k'+1}^{[i]} - |\mathcal{N}_k^{[i]}|}{\delta_{k'+1}^{[i]}}} \prod_{v \in \mathcal{N}_k^{[i]}} (\pi_{k'}^{[v]})^{\frac{1}{\delta_{k'+1}^{[i]}}}. \quad (10)$$

Then for all robots, $\psi_{k'}^{[i]}$ and $\pi_{k'}^{[i]}$ will converge to $\sum_{v=1}^{n_r} \psi_0^{[v]}/n_r$ and $\prod_{v=1}^{n_r} (\pi_0^{[v]})^{1/n_r}$, respectively, in the limit as $k' \rightarrow \infty$.

Proof (Theorem 1). *From Lemma 1, we have for all robots and $k' \geq \text{diam}(\mathcal{G}_k)$ that $\delta_{k'}^{[i]}$ is equal to $(1 + \Delta_k)$. Substituting $(1 + \Delta_k)$ into (9) for all $\delta_{k'}^{[i]}$ and $\delta_{k'+1}^{[i]}$ results in a consensus algorithm that is equivalent to (8) with $\epsilon_k = 1/(1 + \Delta_k)$. In addition, we know for $k' < \text{diam}(\mathcal{G}_k)$ that the time varying nonlinear system (9) will not have worse than exponential divergence since all coefficients in the right hand side of (9) are bounded below and above by 0 and 1, respectively. Thus, since $\psi_{k'}^{[i]}$ in (8) was proven to converge to $\sum_{v=1}^{n_r} \psi_0^{[v]}/n_r$, we have that $\psi_{k'}^{[i]}$ in (9) will do the same if and only if $\sum_{v=1}^{n_r} \psi_{k'}^{[v]}$ equals $\sum_{v=1}^{n_r} \psi_0^{[v]}$ for $k' = \text{diam}(\mathcal{G}_k)$.*

For this aim, consider the summation

$$\sum_{i=1}^{n_r} \delta_{k'+1}^{[i]} \psi_{k'+1}^{[i]} = \sum_{i=1}^{n_r} (\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}) \psi_0^{[i]} + \sum_{i=1}^{n_r} \delta_{k'}^{[i]} \psi_{k'}^{[i]} + \sum_{i=1}^{n_r} \sum_{v \in \mathcal{N}_k^{[i]}} (\psi_{k'}^{[v]} - \psi_{k'}^{[i]})$$

In an undirected graph, the last term on the right hand side of the last equation is equal to zero, and thus we have for all communication rounds that

$$\begin{aligned} \sum_{i=1}^{n_r} \delta_{k'+1}^{[i]} \psi_{k'+1}^{[i]} &= \sum_{i=1}^{n_r} (\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}) \psi_0^{[i]} + \sum_{i=1}^{n_r} (\delta_{k'+1}^{[i]} - \delta_{k'}^{[i]}) \psi_0^{[i]} \\ &+ \dots + \sum_{i=1}^{n_r} (\delta_1^{[i]} - \delta_0^{[i]}) \psi_0^{[i]} + \sum_{i=1}^{n_r} \delta_0^{[i]} \psi_0^{[i]} = \sum_{i=1}^{n_r} \delta_{k'+1}^{[i]} \psi_0^{[i]}, \end{aligned}$$

implying that $\sum_{v=1}^{n_r} \psi_{k'}^{[v]}$ equals $\sum_{v=1}^{n_r} \psi_0^{[v]}$ when $\delta_{k'+1}^{[i]} = \Delta_k$. Thus, we have that $\psi_{k'}^{[i]}$ in (9) converges to $\sum_{v=1}^{n_r} \psi_0^{[v]}/n_r$, which also implies that $\pi_{k'}^{[i]}$ in (10) converges to $\prod_{v=1}^{n_r} (\pi_0^{[v]})^{1/n_r}$. \square

Corollary 2 (Convergence on Complete Network Graphs).

For a complete network graph, $\psi_{k'}^{[i]}$ and $\pi_{k'}^{[i]}$ converge to $\sum_{v=1}^{n_r} \psi_0^{[v]}/n_r$ and $\prod_{v=1}^{n_r} (\pi_0^{[v]})^{1/n_r}$, respectively, after one communication round.

C. Consensus of the Joint Measurement Probabilities

In an earlier paper, we showed that the structuring and eventual consensus of the joint measurement probabilities relied on the indices of the elements in the environment state alphabet, and as a result this alphabet was assumed to be of finite size describing a discrete-valued random variable [11]. This assumption in the previous work was partly motivated by the fact that the robots' beliefs were represented in full, and thus the Bayesian prediction and update calculations required some form of quantization. Here we do not need to

make this assumption for the inference as our non-parametric methods naturally account for continuous distributions.

For continuous distributions well approximated by certain parametric statistics, the consensus-based algorithm can be readily used to average distribution parameters instead of the joint measurement probabilities themselves. For example, the approximation of a joint Gaussian distribution converges using only two parameters for the consensus-based algorithm, and the number of parameters used to represent a multivariate distribution scales quadratically with respect to the distribution's dimension [12]. We also showed that mixtures of Gaussians can be used for “arbitrary” continuous distributions. Nonetheless, in this paper we do assume for the sake of simplicity a finite sized environment state alphabet, $\mathcal{X} = \{x^{[j]} : j \in \{1, \dots, |\mathcal{X}|\}\}$, but note that this assumption is not necessary in general.

For consensus of the sampled joint measurement probabilities, let $\pi_{k'}^{[i]}$ be a belief matrix³ representing the unnormalized approximated n_r th root (i.e., $\sqrt[n_r]{\cdot}$) of these probabilities known by the i th robot after k' communication rounds. In addition, let the belief matrix for all $j \in \{1, \dots, |\mathcal{X}|\}$ and $\ell \in \{1, \dots, n_y\}$ be initialized as $[\pi_0^{[i]}]_{j\ell} = \mathbb{P}(\tilde{Y}_k^{[i]} = \tilde{y}_k^{[i,\ell]} | X_k = x^{[j]})$. In words, the belief matrix is initialized to the i th robot's conditionally independent contribution to the unnormalized sampled joint measurement probabilities,

$$\frac{\mathbb{P}(\tilde{Y}_k^{[i]} | X_k)}{\eta} = \prod_{i=1}^{n_r} \mathbb{P}(\tilde{Y}_k^{[i]} | X_k), \quad (11)$$

where η is a normalization factor.

By allowing the belief matrix to evolve using (10), we define an approximation for the sampled joint measurement probabilities,

$$[\tilde{p}_k^{[i]}]_{j\ell} := \frac{[\pi_k^{[i]}]_{\xi\ell}^{\beta_k^{[i]}}}{\sum_{\ell'=1}^{n_y} [\pi_k^{[i]}]_{\xi\ell'}^{\beta_k^{[i]}}} \approx \mathbb{P}(\tilde{Y}_k = \tilde{y}_k^{[\ell]} | X_k = \tilde{x}_k^{[i,j]}), \quad (12)$$

for all $j \in \{1, \dots, n_x\}$ and $\ell \in \{1, \dots, n_y\}$, where ξ is such that $\tilde{x}_k^{[i,j]} = x^{[\xi]}$, $\pi_k^{[i]}$ is shorthand denoting $\pi_{k'}^{[i]}$ with $k' = n_\pi$, and $\beta_k^{[i]}$ is an exponential factor accounting for the fact that the consensus-based algorithm may terminate before converging. More specifically, $\pi_k^{[i]}$ can be thought of as a weighted logarithmic summation of $\mathbb{P}(\tilde{Y}_k^{[v]} | X_k)$ over all $v \in \{1, \dots, n_r\}$, and $\beta_k^{[i]}$ is the inverse of the largest weight to ensure that no single measurement probability in the right hand side product of (11) has an exponent of value larger than one. In other words, no observation “gets counted” more than once.

To calculate the exponential factor $\beta_k^{[i]}$ in parallel with the belief matrix, let the term $\psi_{k'}^{[i]}$ evolve by using (9) and be initialized to \mathbf{e}_i , where \mathbf{e}_i is the standard basis pointing in the i th direction in \mathbb{R}^{n_r} . From the discussion above, we have that $\beta_k^{[i]} = \|\psi^{[i]}\|_\infty^{-1}$, where $\psi^{[i]}$ is shorthand denoting $\psi_{k'}^{[i]}$ with $k' = n_\pi$. Note that $\psi_0^{[i]}$ does not need to be of size n_r , which would violate the assumption that the number of robots is unknown. Instead, each robot maintains an indexed

vector initially of size one, then augments this vector when unknown indices are received during the consensus round.

From Theorem 1, we have for all robots that $\beta_k^{[i]}$ converges to n_r in the limit as $n_\pi \rightarrow \infty$, or after one communication round if the network is complete (Corollary 2). This property is required for the convergence of the approximations to the true sampled joint measurement probabilities, which will be discussed in Theorem 2. Nonetheless, we showed that the robots can use consensus of the belief matrix (as will be summarized in Algorithm 3) to enable the gradient-based distributed controller from Corollary 1, which continuously moves the robots to improve the informativeness of the next joint observation.

Remark 2 (Complexity of the Exponential Factor). *One should immediately recognize that the computational and network complexity of calculating the exponential factor is linear with respect to the number of robots if the network graph remains complete. For our applications using mobile ad-hoc networks, the finite bandwidth of each robot's communication device results in sparse graphs for large numbers of robots. Once the network becomes fully loaded, the complexity of the exponential factor remains constant with respect to the number of robots.*

Once an observation is received, a second consensus round is performed to update the weighted environment state sample set. Let $\pi_{k'}^{[i]}$ now be a belief vector representing the normalized approximated n_r th root of the observed joint measurement probabilities, $\mathbb{P}(Y_k = y_k | X_k)$, known by the i th robot after k' communication rounds. In addition, let the belief vector for all $j \in \{1, \dots, |\mathcal{X}|\}$ be initialize as $[\pi_0^{[i]}]_j = \mathbb{P}(Y_k^{[i]} = y_k^{[i]} | X_k = x^{[j]})$. After n_π communication rounds, the approximation for the observed joint measurement probabilities is given by

$$[p_k^{[i]}]_j := [\pi_k^{[i]}]_{\xi}^{\beta_k^{[i]}} \approx \mathbb{P}(Y_k = y_k | X_k = \tilde{x}_k^{[i,j]}) \quad (13)$$

for all $j \in \{1, \dots, n_x\}$. The process of forming the approximation is also summarized in Algorithm 3, and the result is used to update the weighted environment state sample set for the next time step $k + 1$.

Lastly, we prove that both distributed approximations, $\tilde{p}_k^{[i]}$ and $p_k^{[i]}$, converge to their corresponding joint measurement probabilities.

Theorem 2. *For all robots, $j \in \{1, \dots, n_x\}$, and $\ell \in \{1, \dots, n_y\}$, we have that $[\tilde{p}_k^{[i]}]_{j\ell}$ and $[p_k^{[i]}]_j$ converge to $\mathbb{P}(\tilde{Y} = \tilde{y}^{[\ell]} | X = x^{[\xi]})$ and $\mathbb{P}(Y | X = x^{[\xi]})$, respectively, in the limit as $n_\pi \rightarrow \infty$, where ξ is again such that $\tilde{x}^{[i,j]} = x^{[\xi]}$. In addition, this convergence happens after one communication round for a complete network graph.*

Proof (Theorem 2). *For all robots, $j \in \{1, \dots, |\mathcal{X}|\}$, and $\ell \in \{1, \dots, n_y\}$, let $[\pi_{k'}^{[i]}]_{j\ell}$ be initialized to $\mathbb{P}(\tilde{Y}_k^{[i]} = \tilde{y}_k^{[i,\ell]} | X_k = x^{[j]})$ and evolve using (10). From Theorem 1, we have that $[\pi_k^{[i]}]_{j\ell}$ and $\beta_k^{[i]}$ converge to $\prod_{v=1}^{n_r} \mathbb{P}(\tilde{Y}_k^{[v]} =$*

³We are using terminology introduced by Pearl in [17]

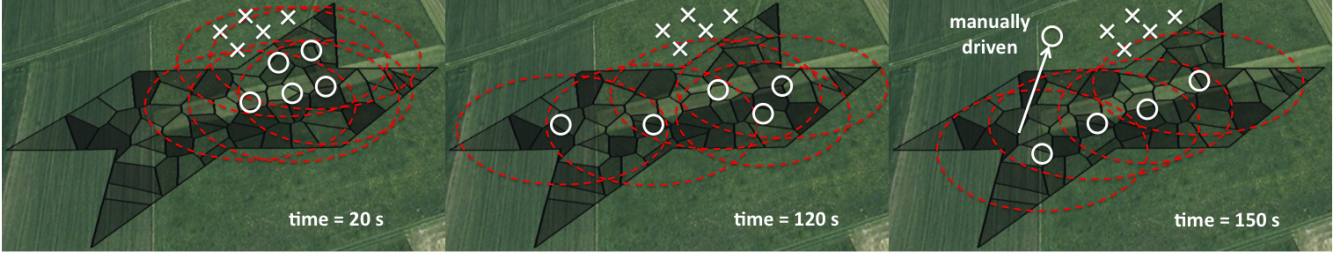


Fig. 2. This figure shows the time evolution of the constellation of five quadrotor flying robots (white circles) with simulated sensor (red dashed circles). These robots are tasked to explore a 150 m wide discretized outdoor environment containing 58 cells of continuous-valued states. The experiment starts with all robots hovering at their starting positions (white \times 's). *Left*: The robots at time = 20 s have begun to explore the environment. *Middle*: At time = 120 s, the distributed controller has driven the robots into a configuration that covers a large portion of the cells. *Right*: One of the robots is manually driven to leave the environment in order to show how the controller adapts to external tasking (or possible robot failures). Thirty seconds later, at time = 150 s, the robots have regrouped to a configuration that positions their sensors to minimize the remaining uncertainty about the environment state.

Algorithm 3 Belief_Consensus(*type*)

```

1:  $\delta_0^{[i]} \leftarrow \mathbf{e}_i$ .
2:  $\psi_0^{[i]} \leftarrow (|\mathcal{N}_k^{[i]}| + 1)$ 
3: if type is sampled then
4:    $[\pi_0^{[i]}]_{j\ell} \leftarrow \mathbb{P}(\tilde{Y}^{[i]} = \tilde{y}^{[i,\ell]} | X = x^{[j]}), \quad \forall j, \ell$ .
5: else if type is observed then
6:    $[\pi_0^{[i]}]_j \leftarrow \mathbb{P}(Y^{[i]} = y_k^{[i]} | X = x^{[j]}), \quad \forall j$ .
7: end if
8: for  $k' = 1$  to  $n_\pi$  do
9:    $\delta_{k'}^{[i]} \leftarrow \max \{ \{ \delta_{k'-1}^{[i]} \} \cup \{ \delta_{k'-1}^{[v]} : v \in \mathcal{N}_k^{[i]} \} \}$ .
10:  Update  $\psi_{k'}^{[i]}$  and  $\pi_{k'}^{[i]}$  using (9) and (10), respectively.
11: end for
12:  $\beta_k^{[i]} \leftarrow \|\psi^{[i]}\|_\infty^{-1}$ .
13: if type is sampled then
14:   return  $\tilde{p}_k^{[i]}$  from (12)
15: else if type is observed then
16:   return  $p_k^{[i]}$  from (13)
17: end if

```

$\tilde{y}^{[v,\ell]} | X_k = x^{[j]})^{1/n_r}$ and n_r , respectively, in the limit as $n_\pi \rightarrow \infty$. Hence from (11) we have for all $j \in \{1, \dots, n_x\}$ that $[\pi^{[i]}]_{j\ell}$ converges to $\mathbb{P}(\tilde{Y} = \tilde{y}^{[i,\ell]} | X_k = x^{[j]})$.

Lastly, using the definition of $\tilde{p}_k^{[i]}$ from (12), we have that $[\tilde{p}_k^{[i]}]_{j\ell}$ converges to $\mathbb{P}(\tilde{Y}_k = \tilde{y}_k^{[i,\ell]} | X = x^{[j]})$. Convergence after one communication round for a complete network graph is a direct consequence of Corollary 2.

The proof for $p_k^{[i]}$ follows in the same manner with $[\pi_{k'}^{[i]}]_j$ being initialized to $\mathbb{P}(Y_k^{[i]} = y_k^{[i]} | X_k = x^{[j]})$ and converging to $\mathbb{P}(Y_k = y_k | X_k = x^{[j]})$. \square

V. EXPERIMENTS IN EXPLORATION

The task for our hardware experiments was to infer the state of a bounded, planar environment by deploying five Ascending Technologies Hummingbird quadrotor flying robots belonging to the class of systems described in Corollary 1. The 150 m wide environment (see Figures 1-3) was discretized using a Voronoi partitioner into 58 heterogeneous cells. We chose this type of partitioning due to its applicability in representing spatial information - note that our

approach does not require any specific type of partitioning. The realtime non-parametric inference and coordination algorithm (Algorithm 1) ran in distributed fashion on a single computer, from which waypoints for each robot were sent to its onboard GPS controller via a 2.4 Ghz Xbee-Pro wireless module.

Five heterogeneous sensors were simulated with measurement noise that was proportional to the sensor radii of (30.0, 32.5, 35.0, 37.5, 40.0) m, which we emphasized by setting the hovering height of a robot proportional to its value. In words, robots hovering closer to the environment had more accurate observations, but also had smaller fields of view. Each robot used a control policy set of $\mathcal{U}^{[i]} = \{[-3 \text{ m/s}, 3 \text{ m/s}]\}^2$ and a control gain of $\gamma^{[i]} = 5$, while sample sizes of $n_x = n_y = 500$ allowed for a sampling interval of $T_s = 1$. Finally, the consensus-based algorithm of round sizes $n_\pi = 3$ was implemented on the undirected network graph using an ideal disk model of radius 50 m to determine connectivity.

Reproducible results were acquired in two separate experiment setups, producing over 25 minutes of total flight time. The first setup implemented the non-parametric methods without any high-level control except for the manual override capabilities enabled by the Disaster Management Tool (DMT) developed at DLR [7]. The robots were deployed from outside the nonconvex environment, and for each environment cell, the random variables for the state and the corresponding robot cell observation took continuous values between 0 and 1 (i.e., a generic representation for the continuous domain). Figure 2 shows the robots being deployed, expanding over the environment, and automatically compensating for a robot that was manually tasked to leave the environment. The robots continuously adjusted their configuration to address areas of high uncertainty, a behavior that is the result of not having enough sensing capability to statically cover the environment.

The qualitative performance of the initial experiments acted as a proof of concept for the approach, leading to the integration into a larger exploration system (Figure 3). The algorithms for the second experiment setup were adjusted

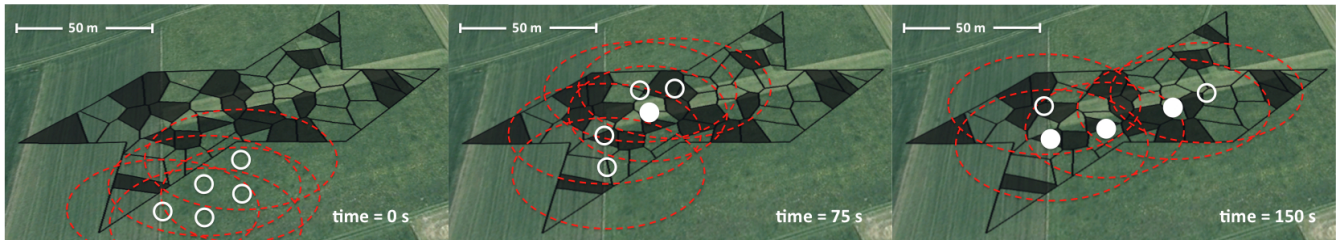


Fig. 3. This figure shows the constellation exploring the environment containing cells of binary state. *Left*: The experiment starts with all robots hovering at their starting positions. *Middle*: The robots at time = 75 s have begun to explore the environment. In addition, one robot is assigned by a higher level communication scheme to act as a dynamic communication relay (white filled circle), and thus the control actions produced by the distributed controller are overridden for that robot. *Right*: At time = 150 s, even though three robots are assigned as dynamic relays, the distributed controller has driven the system into a configuration that covers a large portion of the cells.

to handle binary event detection (e.g., fire or no fire), and a high-level communication scheme was implemented to continuously assign robots to act as dynamic communication relays. The robots were deployed in a similar fashion from outside the environment, and at any given point could have at most 58 bits of uncertainty concerning the inference. Figure 1 shows the decrease in entropy over the extent of the experiment, even though at times the communication scheme was overriding the inputs from the distributed controller.

VI. CONCLUSIONS

We presented a novel suite of representations and methods for distributed inference and coordination in multi-robot systems. In contrast to previous approaches, no underlying assumptions are needed concerning the type of probability distributions present in the system, nor what knowledge the robots have about the topology of the communication network. Nevertheless, the overall complexity in terms of computation, memory, and network load remains constant with respect to number of robots, resulting in a scalable yet robust approach for information acquisition tasks. To validate this concept, we conducted large-scale outdoor experiments that showed fully autonomous exploration of a bounded environment with five quadrotor flying robots. Moreover, we demonstrated how the algorithm automatically adapts to manual overrides and high-level control inputs. We believe the ability to combine low-level autonomy with hierarchical cognitive supervision is particularly advantageous and an important step towards fieldable systems in the foreseeable future.

REFERENCES

- [1] J. Borenstein, L. Feng, and H. R. Everett. *Navigating Mobile Robots: Systems and Techniques*. Data Management Systems Series. Morgan Kaufmann Publishers, Inc., 1997.
- [2] F. Bourgault, A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte. Information based adaptive robotic exploration. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 540–545, 2002.
- [3] A. Cameron and H. Durrant-Whyte. A bayesian approach to optimal sensor placement. *The International Journal of Robotics Research*, 9(5):70, 1990.
- [4] J. Cortés. Distributed kriged Kalman filter for spatial estimation. *IEEE Transactions on Automatic Control*, 54(12):2816–2827, 2009.
- [5] S. P. Engelson and D. V. McDermott. Error correction in mobile robot map learning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 2555–2560, May 1992.
- [6] D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous Robots*, 2000.
- [7] M. Frassl, M. Lichtenstern, M. Khider, and M. Angermann. Developing a system for information management in disaster relief - methodology and requirements. In *Proceedings of the International Community on Information Systems for Crisis Response and Management Conference*, Seattle, USA, May 2010.
- [8] B. Grocholsky. *Information-theoretic control of multiple sensor platforms*. PhD thesis, University of Sydney, 2002.
- [9] G. M. Hoffmann and C. J. Tomlin. Mobile sensor network control using mutual information methods and particle filters. *IEEE Transactions on Automatic Control*, 55(1):32–47, January 2010.
- [10] A. T. Ihler, J. W. Fischer III, R. L. Moses, and A. S. Willsky. Non-parametric belief propagation for self-localization of sensor networks. *IEEE Journal on Selected Areas in Communications*, 23(4):809–819, April 2005.
- [11] B. J. Julian, M. Angermann, M. Schwager, and D. Rus. A scalable information theoretic approach to distributed robot coordination. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5187–5194, 2011.
- [12] B. J. Julian, S. L. Smith, and D. Rus. Distributed approximation of joint measurement distributions using mixtures of gaussians. In *Proceedings of the Robotics: Science and Systems Conference*, Sydney, Australia, July 2012. Submitted.
- [13] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman. Decentralized environmental modeling by mobile sensor networks. *IEEE Transactions on Robotics*, 24(3):710–724, June 2008.
- [14] N. A. Lynch. *Distributed Algorithms*. Data Management Systems Series. Morgan Kaufmann Publishers, Inc., 1997.
- [15] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 593–598, 2002.
- [16] R. Olfati-Saber, E. Franco, E. Frazzoli, and J. S. Shamma. Belief consensus and distributed hypothesis testing in sensor networks. In *Proceedings of the Network Embedded Sensing and Control Workshop*, pages 169–182, 2005.
- [17] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., 1988.
- [18] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1650–1670, May 2001.
- [19] M. Schwager, P. Dames, D. Rus, and V. Kumar. A multi-robot control policy for information gathering in the presence of unknown hazards. In *Proceedings of the International Symposium on Robotics Research*, Flagstaff, AZ, USA, August 2011.
- [20] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents Series. The MIT Press, 2005.
- [21] L. Xiao, S. Boyd, and S.-J. Kim. Distributed average consensus with least-mean-square deviation. *Journal of Parallel and Distributed Computing*, 67:33–46, January 2007.